synchronization "to determine whether a record of the second database is identical to the first record."

In Salkewicz, a process known as "hash buckets" is used to assign the records of a database to segments, so that one segment at a time can be synchronized. This is an alternative to simply assigning the records sequentially. To assign the records using hash buckets, modulo-3 arithmetic is performed on the record number (not the content) of each record, and records are assigned to a hash bucket based on the outcome of that arithmetic. In the example shown (FIGS. 9A-9C), the result is that the first segment contains records 6, 12, and 9, whereas with the sequential assignment technique (FIGS. 8A-8F), the first segment contains records 1, 2, and 4. Thus, at most, the hash process merely selects the segment into which a record is to be placed.

Thus, in at least three enormous respects, Salkewicz falls short of the claims. First, Salkewicz does not compute a hash number based on at least a portion of the content of a record. In Salkewicz, the modulo-3 arithmetic is applied to the record number of a database record, not to its content. Thus, the number produced by the modulo-3 arithmetic is not based on at least a portion of the content of the record.

Second, Salkewicz does not use the number produced by the modulo-3 arithmetic to determine whether the database record is identical to another record, as part of the synchronization process. Salkewicz uses the outcome of the modulo-3 arithmetic operation merely to determine the segment to which the record is to be assigned. That has nothing to do with a synchronization process, let alone determining whether the record represented by the hash number is identical to another record.

Third, Salkewicz does not compute a hash number that is "insufficient to reconstruct the record but sufficient to identify the record." Clearly, the outcome of the modulo-3 arithmetic is not sufficient to identify a record in Salkewicz, because a plurality of records produce the same modulo-3 outcome, as a plurality of records get assigned to the same hash bucket.

The examiner adds Preneel to Salkewicz, but Preneel adds nothing of relevance to the invention other than the widely known idea that a computer record can be represented by a hash key that is considerably smaller than the record. The examiner agrees with this assessment of Preneel (see page 3, lines 7-16 of the office action).

The invention's use of a content-based hash number as part of a synchronization process, to determine whether a record of one database is identical to a record of another database, is not taught in Salkewicz or Preneel, whether examined separately or together. Salkewicz uses a modulo-3 arithmetic operation to assign database records to segments – an entirely different purpose than the hash operation called for by the claims. Preneel teaches the basics of hash numbers, but nothing about using them in connection with synchronizing database records.

Accordingly, independent claims 35 and 37 are allowable.

The remaining claims are all properly dependent on one or more of the independent claims, and thus allowable therewith. Each of the dependent claims adds one or more further limitations that enhance patentability, but those limitations are not presently relied upon. For that reason, and not because applicants agree with the examiner, no rebuttal is offered to the examiner's reasons for rejecting the dependent claims.

Allowance of the application is requested.

Please apply any other charges or credits to deposit account 06-1050.

Respectfully submitted,

Date: 5-1-06

G. Roger Lee Reg. No. 34,053
G. Roger Lee
Reg. No. 28,963

Fish & Richardson P.C.
225 Franklin Street
Boston, MA 02110
Telephone: (617) 542-5070
Facsimile: (617) 542-8906